Learning in neural networks and an integrable system (Applied Mathematics of Discrete Integrable Systems)

AUTHOR(S):

Fukumizu, Kenji

数理解析研究所講究録
1098巻 1999年 23-41

可積分系によるニューラルネットワークの学習の解析

Learning in neural networks and an integrable system

福水 健次
理化学研究所

Kenji Fukumizu
The Institute of Physical and Chemical Research (RIKEN)
E-mail: fuku@brain.riken.go.jp

Abstract

This paper investigates the dynamics of batch learning of multilayer neural networks in the asymptotic case where the number of training data is much larger than the number of parameters. First, we present experimental results on the behavior in the steepest descent learning of multilayer perceptrons and three-layer linear neural networks. We see in these results that strong overtraining, which is the increase of generalization error in training, occurs if the model has surplus hidden units to realize the target function. Next, to analyze overtraining from the theoretical viewpoint, we analyze the steepest descent learning equation of a three-layer linear neural network, and theoretically show that a network with surplus hidden units presents overtraining. From this theoretical analysis, we can see that overtraining is not a feature observed in the final stage of learning, but it occurs in an intermediate time interval.

1 Introduction

This paper discusses the dynamics of batch learning in neural networks. Multilayer networks like multilayer perceptrons have been extensively used in many engineering applications, in hope that their nonlinear structure inspired by biological neural networks shows a variety of advantages. However, nonlinearity causes weaknesses also. For the training of a neural network, for example, a numerical optimization method like the error back-propagation (Rumelhart et al., 1986) must be used to find the optimal weight and bias parameters. It is very important from the practical and theoretical viewpoints to elucidate the dynamical behavior of a network in training. Especially, the *empirical error*, the objective function of minimization, and the *generalization error*, the difference between the target function and its estimate, are of much interest in many researches. However, the property of such learning rules have not been clarified completely because of its complex nonlinearity.

In this paper, we focus on *overtraining* (Amari, 1996) as a typical behavior of learning in multilayer networks. Overtraining is the increase of the generalization error after some

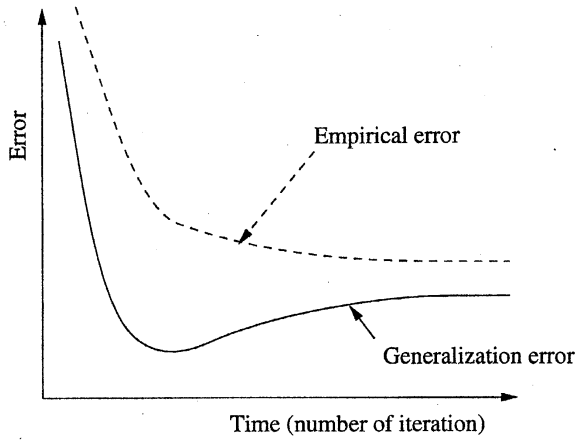

Figure 1: Overtraining in batch learning. The empirical error decreases during iterative training, while the generalization increases after some time point.

time point in learning (Fig.1). Although the empirical error is a decreasing function in principle, it does not ensure the decrease of the generalization error defined by only finite number of training examples. Overtraining is not only of theoretical interest but of practical importance, because we should use an early stopping technique if overtraining really exists.

There has been a controversy about the existence of overtraining. Some practitioners assert that overtraining often happens and propose early stopping methods. Amari et al. (1996) analyze overtraining theoretically and conclude that the effect of overtraining is much smaller than what is believed by practitioners. Their analysis is true if the parameter approaches to the global minimum of the empirical error following the statistical asymptotic theory (Cramér, 1946). However, the usual asymptotic theory cannot be applied in *overrealizable* cases, where the model has surplus hidden units to realize the target function (Fukumizu, 1996; Fukumizu, 1997). Possibility of an overrealizable target is a common property of multilayer models, and the existence of overtraining in such cases has still been an open problem.

There have been other studies related to overtraining. Baldi and Chauvin (Baldi & Chauvin, 1991) theoretically analyze the generalization error of two-layer linear networks estimating the identity function from noisy data. Although they show overtraining phenomena under several assumptions, their analysis is very limited in that the overtraining can be observed only when the variance of the noise in the training sample is different from that of the validation sample. In researches from the viewpoint of statistical mechanics, overtraining is sometimes used for a different meaning. They observe the increase of the generalization error in online learning when the number of data is smaller than the number

of parameters, and call it overtraining.

The main purpose of this paper is to discuss overtraining of multilayer networks in connection with overrealizability. We investigate two models: one is multilayer perceptrons with the sigmoidal activation function, and the other is linear neural networks, which we introduce for the simplicity of analysis. Through experimental results of these models, we can conjecture that overtraining occurs for overrealizable targets. We theoretically verify this conjecture in linear neural networks.

This paper is organized as follows. Section II gives basic definitions and terminology. Section III shows experimental results concerning overtraining of multilayer perceptrons and linear neural networks. In Section IV, we theoretically show the existence of overtraining in overrealizable cases of linear neural networks. Section V includes concluding remarks.

2 Preliminaries

2.1 Framework of statistical learning

In this section, we present basic definitions and terminologies used in this paper. A feedforward neural network model can be defined by a parametric family of maps $\{\boldsymbol{f}(\boldsymbol{x};\boldsymbol{\theta})\}$ from $\mathbb{R}^L$ to $\mathbb{R}^M$, where $\boldsymbol{x}$ is an input vector and $\boldsymbol{\theta}$ is a parameter vector. The three-layer network model with H hidden units is defined by

$$f^i(\boldsymbol{x};\boldsymbol{\theta}) = \sum_{j=1}^{H} w_{ij}\,\varphi\left(\sum_{k=1}^{L} u_{jk}x_k + \zeta_j\right) + \eta_i, \quad (1 \le i \le M) \tag{1}$$

where $\boldsymbol{\theta} = (w_{11},\dots,w_{MH},\eta_1,\dots,\eta_M,u_{11},\dots,u_{HL},\zeta_1,\dots,\zeta_H)$, and an activation function $\varphi(t)$ is a function from $\mathbb{R}$ to $\mathbb{R}$. We call a three-layer network a *multilayer perceptron*, if the sigmoidal function

$$\varphi(t) = \frac{1}{1+e^{-t}} \tag{2}$$

is used for the activation function.

We use such a model for regression problems, assuming that an output of the target system is observed with a measurement noise. A sample $(\boldsymbol{x},\boldsymbol{y})$ from the target system satisfies

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{v}, \tag{3}$$

where $\boldsymbol{f}(\boldsymbol{x})$ is the *target function* which is unknown to a learner, and $\boldsymbol{v}$ is a random vector with 0 as its mean and $\sigma^2 I_M$ as its variance-covariance matrix. We write I_M for the

$M \times M$ unit matrix. An input vector $\boldsymbol{x}$ is generated randomly with its probability density function $q(\boldsymbol{x})$, which is unknown to a learner. Training data $\{(\boldsymbol{x}^{(\nu)}, \boldsymbol{y}^{(\nu)}) | \nu = 1, \ldots, N\}$ are independent samples from the joint distribution of $q(\boldsymbol{x})$ and eq.(3). The parameter $\boldsymbol{\theta}$ is estimated based on the training data without knowing $\boldsymbol{f}(\boldsymbol{x})$ so that a neural network can give a good estimate of the target function $\boldsymbol{f}(\boldsymbol{x})$.

We discuss the least square error (LSE) estimator. The objective of training is to find the parameter that minimizes the following *empirical error*:

$$\mathrm{E}_{emp}(\boldsymbol{\theta}) \equiv \sum_{\nu=1}^{N} \|\boldsymbol{y}^{(\nu)} - \boldsymbol{f}(\boldsymbol{x}^{(\nu)}; \boldsymbol{\theta})\|^2. \tag{4}$$

Generally, it is very difficult to solve the LSE estimator analytically if the model is non-linear. Some numerical optimization method is needed to obtain an approximation. One widely-used method is the steepest descent method, which iteratively updates the parameter vector $\boldsymbol{\theta}$ in the steepest descent direction of the error surface defined by $\mathrm{E}_{emp}(\boldsymbol{\theta})$. The learning rule is given by

$$\boldsymbol{\theta}(t+1) = \boldsymbol{\theta}(t) - \beta \frac{\partial \mathrm{E}_{emp}(\boldsymbol{\theta}(t))}{\partial \boldsymbol{\theta}}, \tag{5}$$

where $\beta > 0$ is a learning constant. In this paper, we discuss only batch learning, in which we calculate the gradient using all the training data. There are many researches on online learning, in which the parameter is always updated for a newly generated data (Heskes & Kappen, 1991).

The performance of a network is often evaluated by the *generalization error*, which represents the average error between the target function and its estimate. More precisely, it is defined by

$$\mathrm{E}_{gen}(\boldsymbol{\theta}) \equiv \int \|\boldsymbol{f}(\boldsymbol{x}; \boldsymbol{\theta}) - \boldsymbol{f}(\boldsymbol{x})\|^2 q(\boldsymbol{x}) d\boldsymbol{x}. \tag{6}$$

It is easy to see

$$\int\int \|\boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}; \boldsymbol{\theta})\|^2 p(\boldsymbol{y}|\boldsymbol{x}) q(\boldsymbol{x}) d\boldsymbol{y} d\boldsymbol{x} = M\sigma^2 + \mathrm{E}_{gen}, \tag{7}$$

where $p(\boldsymbol{y}|\boldsymbol{x})$ is the density function of the conditional probability of $\boldsymbol{y}$ given $\boldsymbol{x}$. Since the empirical error divided by N approaches to the left hand side of eq.(7) when N goes to infinity, the minimization of the empirical error roughly approximates the minimization of the generalization error. However, they are not exactly the same, and the decrease of the empirical error during learning does not ensure the decrease of the generalization error. It is extremely important to elucidate the dynamical behavior of the generalization error. A curve showing the empirical/generalization error as a function of time is called a learning curve.

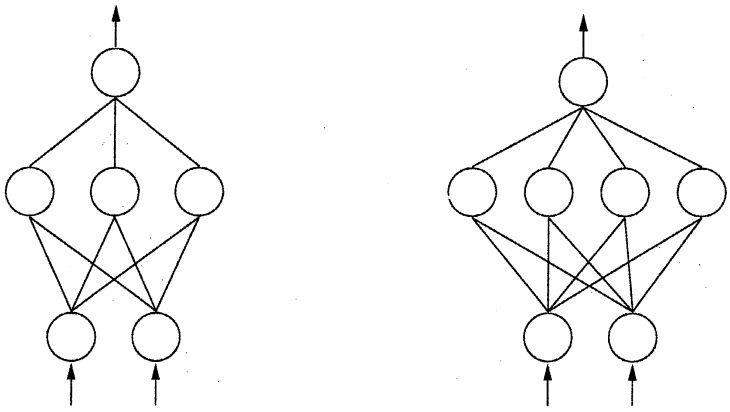

Figure 2: Illustration of an overrealizable case.

In our theoretical discussion, we consider the case where $\boldsymbol{f}(\boldsymbol{x})$ is perfectly realized by the prepared model; that is, there is a true parameter $\boldsymbol{\theta}_0$ such that $\boldsymbol{f}(\boldsymbol{x};\boldsymbol{\theta}_0) = \boldsymbol{f}(\boldsymbol{x})$. Assume that the model has H hidden units. If the target function $\boldsymbol{f}(\boldsymbol{x})$ is realized by a network with a smaller number of hidden units than H, we call it *overrealizable* (see Fig.2). Otherwise, we call it *regular*. We focus on the difference of dynamical behaviors between these two cases.

2.2 Linear neural networks

We introduce the three-layer linear neural network model as the simplest multilayer model whose theoretical analysis is possible. We must be very careful in discussing experimental results on learning of a multilayer perceptron especially in an overrealizable case. In such a case, around the global minimum there exists a high-dimensional subvariety on which E_{emp} is very close to the minimum (Fukumizu, 1997). The convergence of learning is much slower than the convergence in regular cases because the gradient is very small around this subvariety. In addition, of course, learning with a gradient method suffers from local minima, which is a common problem to all nonlinear models. We cannot exclude their effects, and this often makes derived conclusions obscure. On the other hand, the theoretical analysis of linear neural networks is possible (Baldi & Hornik, 1995; Fukumizu, 1997), and the dynamical behavior of the generalization error is analyzed in Section 4.

A linear neural network has the identity function for the activation function. In this paper, we do not use the bias terms $\boldsymbol{\eta}$ and $\boldsymbol{\zeta}$ in linear neural networks for simplicity. Thus,

a linear neural network is defined by

$$\boldsymbol{f}(\boldsymbol{x}; A, B) = BA\boldsymbol{x}, \tag{8}$$

where A is a $H \times L$ matrix and B is a $M \times H$ matrix. We assume $H \leq L$ and $H \leq M$ throughout this paper. Although the function $\boldsymbol{f}(\boldsymbol{x}; A, B)$ is linear, from the assumption on H, the model is not equal to the set of all the linear maps from $\mathbb{R}^L$ to $\mathbb{R}^M$, but is the set of the linear maps of rank no greater than H. Then, it is not equivalent to the usual linear regression model $\boldsymbol{f}(\boldsymbol{x}; C) = C\boldsymbol{x}$ ($C : M \times L$ matrix). In this sense, the three-layer linear neural network model is the simplest multilayer model.

3 Learning curves – experimental study –

In this section, we experimentally investigate the generalization error of multilayer perceptrons and linear neural networks to see their actual behaviors, emphasizing on overtraining.

The steepest descent method (eq.(5)) in multilayer perceptron leads the well-known error back-propagation rule (Rumelhart, 1986), which is given by

$$\begin{aligned}
w_{ij}(t+1) &= w_{ij}(t) + \beta \textstyle\sum_{\nu=1}^{N} \delta_i^{(\nu)}(t) s_j^{(\nu)}(t), \\
\eta_i(t+1) &= \eta_i(t) + \beta \textstyle\sum_{\nu=1}^{N} \delta_i^{(\nu)}(t), \\
u_{jk}(t+1) &= u_{jk}(t) + \beta \textstyle\sum_{\nu=1}^{N} \sum_{i=1}^{M} w_{ij} \delta_i^{(\nu)}(t) s_j^{(\nu)}(t)(1 - s_j^{(\nu)}(t)) x_k^{(\nu)}, \\
\zeta_j(t+1) &= \zeta_j(t) + \beta \textstyle\sum_{\nu=1}^{N} \sum_{i=1}^{M} w_{ij} \delta_i^{(\nu)}(t) s_j^{(\nu)}(t)(1 - s_j^{(\nu)}(t)),
\end{aligned} \tag{9}$$

where

$$\delta_i^{(\nu)}(t) = y_i^{(\nu)} - f_i(\boldsymbol{x}^{(\nu)}; \boldsymbol{\theta}(t)), \quad s_j^{(\nu)}(t) = s\left(\textstyle\sum_{k=1}^{L} u_{jk}(t) x_k^{(\nu)} + \zeta_j(t)\right). \tag{10}$$

To avoid the problems discussed in Section 2.2 as much as possible, we adopt the following configuration in our experiments. The model in use has 1 input unit, 2 hidden units, and 1 output unit. For a fixed set of training examples, 30 different values are tried for an initial parameter vector. We select the trial that gives the least empirical error after 500000 iterations. Figure 3 shows the average of generalization errors over 30 different simulations changing the set of training data. It shows clear overtraining in the overrealizable case. On the other hand, the learning curve in the regular case shows no meaningful overtraining.

It is known that the LSE estimator of the linear neural network model can be analytically solvable (Baldi & Hornik, 1995). Although it is practically absurd, of course, to use the steepest descent method, our interest is not only the optimal estimator but the dynamics of learning. Therefore, we perform the steepest descent learning of linear neural networks

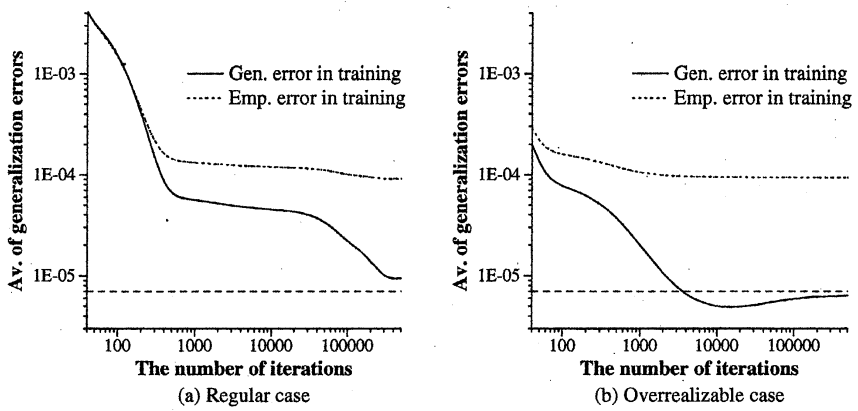

Figure 3: Average learning curves of MLP. The input samples are independent samples from $N(0,4)$, and the output samples include the observation noise subject to $N(0,10^{-4})$. The total number of training data is 100. The constant zero function is used for the overrealizable target function, and $f(x) = s(x+1) + s(x-1)$ is used for the regular target. The dotted line shows the theoretical expectation of the generalization error of the LSE estimator when the target function is regular and the usual asymptotic theory is applicable.

here. For the training data of a linear neural network, we use the notations;

$$X = \begin{pmatrix} \boldsymbol{x}^{(1)T} \\ \vdots \\ \boldsymbol{x}^{(N)T} \end{pmatrix}, \qquad Y = \begin{pmatrix} \boldsymbol{y}^{(1)T} \\ \vdots \\ \boldsymbol{y}^{(N)T} \end{pmatrix}. \tag{11}$$

The empirical error can be written by

$$\mathrm{E}_{emp} = \mathrm{Tr}[(Y - XA^TB^T)^T(Y - XA^TB^T)]. \tag{12}$$

The learning rule of a linear neural network is given by

$$\begin{cases} A(t+1) &= A(t) + \beta\{B^TY^TX - B^TBAX^TX\}, \\ B(t+1) &= B(t) + \beta\{Y^TXA^T - BAX^TXA^T\}. \end{cases} \tag{13}$$

It is also known that all the critical points but the global minimum of E_{emp} are saddle points (Baldi & Hornik, 1995). Therefore, if we theoretically consider a continuous-time learning equation, it converges to the global minimum for almost all initial conditions. Figure 4 shows the average of learning curves for 100 simulations with various training data from the same probability. The number of input units, hidden units, and output units of the model are 5, 3, and 5 respectively. All the overrealizable cases show clear overtraining. while the regular case does not show meaningful overtraining. We can also see that overtraining is stronger when the rank of the target is smaller.

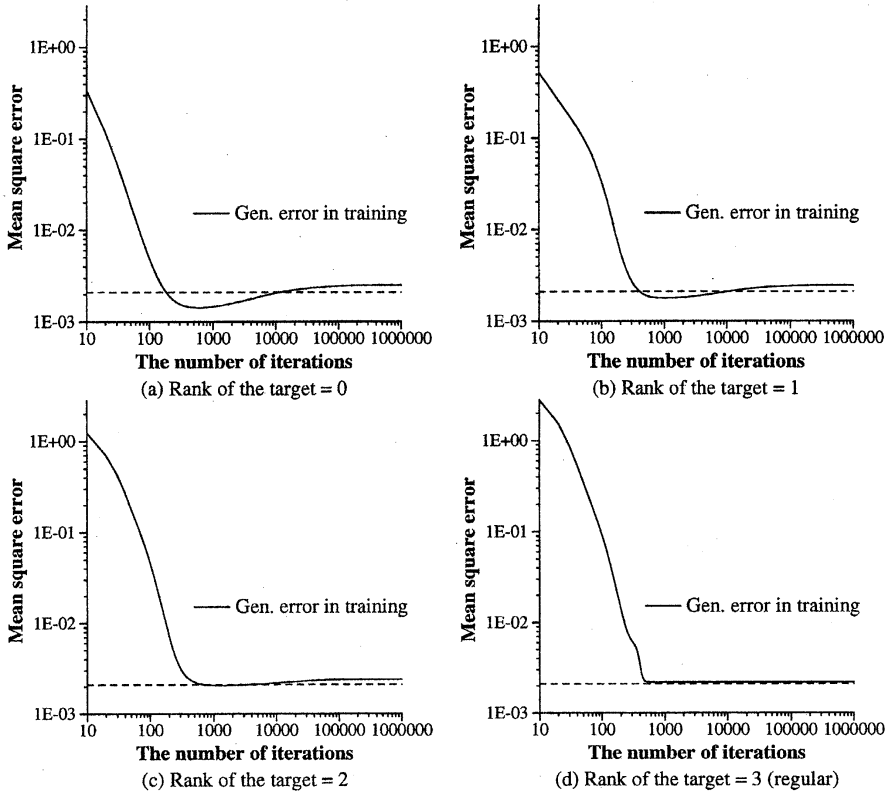

Figure 4: Average learning curves of linear neural networks. The input samples are independent samples from $N(0, I_5)$, and the output samples include the observation noise subject to $N(0, 10^{-2})$. The total number of training data is 100. The dotted line shows the theoretical expectation of the generalization error of the LSE estimator for the regular target.

From these results, we can conjecture that there is an essential difference in the dynamics of learning between regular and overrealizable cases, and overtraining is a universal property of the latter cases. If we use a good stopping criterion, the multilayer networks may have an advantage over conventional linear models, in that the degrade of the generalization error by redundant parameters is not so critical as conventional linear models. Such overtraining in overrealizable cases has not been explained theoretically yet. In the next section, we theoretically verify the result in linear neural networks by obtaining the solution of the steepest descent learning equation.

4 Solvable dynamics of learning in linear neural networks

To verify the existence of overtraining in linear neural networks, we analyze the continuous-time differential equation instead of the discrete-time learning rule. Although the behavior of the continuous one and the discrete one are not identical, they show very similar qualitative behaviors in computer simulations. We further approximate the differential equation to derive a solvable one. This approximation also affects the solution quantitatively. However, we can see in experiments that similar the original one and the approximated one show similar properties in their learning curves, and we can show the existence of overtraining in overrealizable cases.

4.1 Solution of learning dynamics

In the rest of the paper, we put the following assumptions;

(a) $H \leq L = M$,

(b) $\boldsymbol{f}(\boldsymbol{x}) = B_0 A_0 \boldsymbol{x}$,

(c) $\mathrm{E}[\boldsymbol{x}\boldsymbol{x}^T] = \tau^2 I_L$,

(d) $A(0)A(0)^T = B(0)^T B(0)$,

(e) The rank of $A(0)$ (or, equivalently under (d), the rank of $B(0)$) is H.

Under these assumptions, A^T and B are $L \times H$ matrixes. The parameterization of a linear neural network has H^2 dimensional redundancy by the multiplication of $H \times H$ nonsingular matrix from the left of A and its inverse from the right of B. The initial condition (d) does not restrict possible initialization, since it reduces this redundancy with $\frac{1}{2}H(H+1)$ restrictions. The assumption (e) is important in the following discussion. If the rank of $A(0)$ (or $B(0)$) is less than H, the final state of the variables changes, while we do not discuss it in this paper.

If we divide the output as

$$Y = XA_0^T B_0^T + V, \tag{14}$$

the differential equation of the steepest descent learning is given by

$$\begin{cases} \dot{A} &= \beta(B^T B_0 A_0 X^T X + B^T V^T X - B^T B A X^T X), \\ \dot{B} &= \beta(B_0 A_0 X^T X A^T + V^T X A^T - B A X^T X A^T). \end{cases} \tag{15}$$

This is a nonlinear differential equation which has the terms of the third order. Let $Z_O := \frac{1}{\sigma} V^T X (X^T X)^{-1/2}$. It is easy to see that Z_O is independent of X, and that all the elements of Z_O are mutually independent and subject to $N(0,1)$. We decompose $X^T X$ as

$$X^T X = \tau^2 N I_L + \tau^2 \sqrt{N} Z_I, \tag{16}$$

where the off-diagonal elements of Z_I are subject to $N(0,1)$ and the diagonal elements are subject to $N(0,2)$ if N goes to infinity. Let

$$F = B_0 A_0 + \frac{1}{\sqrt{N}}(B_0 A_0 Z_I + \frac{\sigma}{\tau} Z_O). \tag{17}$$

We neglect the order $O(\sqrt{N})$ in the nonlinear terms to derive a solvable equation, and approximate eq.(15) by

$$\begin{cases} \dot{A} &= \beta\tau^2 N B^T F - \beta\tau^2 N B^T B A, \\ \dot{B} &= \beta\tau^2 N F A^T - \beta\tau^2 N B A A^T. \end{cases} \tag{18}$$

Eq.(18) is a good approximation of the original eq.(15) if N is very large.

We further put an assumption:

(f) The rank of $B(0) + FA(0)^T$ is H.

This assumption is a technical one, which is used to prove overtraining. The assumption (f) is almost always satisfied if the initial condition is independent of F.

We have $AA^T = B^T B$ because of the fact $\frac{d}{dt}(AA^T) = \frac{d}{dt}(B^T B)$ and the assumption (d). If we introduce $2L \times H$ matrix

$$R = \begin{pmatrix} A^T \\ B \end{pmatrix}, \tag{19}$$

then, R satisfies the differential equation

$$\frac{dR}{dt} = \beta\tau^2 N S R - \frac{\beta\tau^2 N}{2} R R^T R, \qquad \text{where} \quad S = \begin{pmatrix} 0 & F^T \\ F & 0 \end{pmatrix}. \tag{20}$$

This is very similar to Oja's learning equation (Oja, 1989; Yan et al., 1994), or the generalized Rayleigh quotient (Helmke & Moore, 1994), which is known to be a solvable nonlinear differential equation (Yan et al., 1994).

The key fact to solve eq.(20) is to derive the differential equation on RR^T:

$$\frac{d}{dt}(RR^T) = \beta\tau^2 N S R R^T + \beta\tau^2 N R R^T S - \beta\tau^2 N (R R^T)^2. \tag{21}$$

This is a matrix Riccati differential equation and well-known to be solvable. We have the following

Theorem 1. *Assume that the rank of $R(0)$ is full. Then, the Riccati differential equation (21) has a unique solution for all $t \geq 0$, and the solution is given by*

$$R(t)R^T(t) = e^{\beta\tau^2 N S t} R(0) \Big[I_H + \frac{1}{2} R(0)^T \{ e^{\beta\tau^2 N S t} S^{-1} e^{\beta\tau^2 N S t} - S^{-1} \} R(0) \Big]^{-1} R(0)^T e^{\beta\tau^2 N S t}. \tag{22}$$

For the proof, see Sasagawa (1982) and Yan et al. (1994). Using this theorem, we easily obtain the following theorem in the same manner as Yan et al. (1994, Theorem 2.1).

Theorem 2. *Assume that the rank of $R(0)$ is full. Then, the differential equation (20) has a unique solution for all $t \geq 0$, and the solution is expressed as*

$$R(t) = e^{\beta\tau^2 NSt}R(0)\Big[I_H + \frac{1}{2}R(0)^T\{e^{\beta\tau^2 NSt}S^{-1}e^{\beta\tau^2 NSt} - S^{-1}\}R(0)\Big]^{-\frac{1}{2}}U(t), \tag{23}$$

where $U(t)$ is a $H \times H$ orthogonal matrix.

4.2 Dynamical behavior of the generalization error

In this subsection, we write $M(l \times n; \mathbb{R})$ for the set of real $l \times n$ matrixes. Using the assumption (c), the generalization error is written as

$$\mathrm{E}_{gen} = \tau^2 \mathrm{Tr}[(BA - B_0A_0)(BA - B_0A_0)^T]. \tag{24}$$

It is easy to see that the derivative of E_{gen} is

$$\frac{d}{dt}\mathrm{E}_{gen} = \beta\tau^4 N\left\{\mathrm{Tr}\big[(F - BA)A^TA(BA - B_0A_0)^T\big] + \mathrm{Tr}\big[BB^T(F - BA)(BA - B_0A_0)^T\big]\right\}. \tag{25}$$

Note that a transform of the input or output vectors by an orthogonal matrix does not change the generalization error. Therefore, using the singular value decomposition of B_0A_0, we can assume without loss of generality that B_0A_0 is diagonal;

$$B_0A_0 = \begin{pmatrix} \Lambda^{(0)} & 0 \\ 0 & 0 \end{pmatrix}, \qquad \Lambda^{(0)} = \begin{pmatrix} \lambda_1^{(0)} & & 0 \\ & \ddots & \\ 0 & & \lambda_r^{(0)} \end{pmatrix}, \tag{26}$$

where r is the rank of B_0A_0, and $\lambda_1^{(0)} \geq \cdots \geq \lambda_r^{(0)} > 0$ are the singular values of B_0A_0. The rank r is equal to or less than H. The target function is overrealizable if and only if $r < H$.

We employ the singular value decomposition of F;

$$F = W\Lambda U^T. \tag{27}$$

In this equation, U and W are L dimensional orthogonal matrixes, and

$$\Lambda = \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_L \end{pmatrix}, \tag{28}$$

where $\lambda_1 \geq \lambda_2 \geq \cdots \lambda_L \geq 0$ are the singular values of F. We assume that $\lambda_1 > \lambda_2 > \cdots > \lambda_L > 0$. This is satisfied almost surely because of the stochastic noise in F. We write, for simplicity,

$$F = B_0 A_0 + \varepsilon Z, \tag{29}$$

where $\varepsilon = \frac{1}{\sqrt{N}}$. We further put an assumption:

(g) $\lambda_r^{(0)} \gg \varepsilon$ and $1 \gg \varepsilon$.

The above assumption asserts that the number of data is large enough to discriminate between $\lambda_r^{(0)}$ and the stochastic deviation from zero. We say a is of the constant order if $a \gg \varepsilon$.

It is known that a small perturbation of a matrix causes a perturbation of the same order to the singular values (see Bartia, 1997, Section III, for example). The diagonal matrix Λ is decomposed as

$$\Lambda = \begin{pmatrix} \Lambda_1 & & 0 \\ & \varepsilon\tilde{\Lambda}_2 & \\ 0 & & \varepsilon\tilde{\Lambda}_3 \end{pmatrix}, \tag{30}$$

where

$$\Lambda_1 = \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_r \end{pmatrix} = \begin{pmatrix} \lambda_1^{(0)} + O(\varepsilon) & & 0 \\ & \ddots & \\ 0 & & \lambda_r^{(0)} + O(\varepsilon) \end{pmatrix}, \tag{31}$$

$$\varepsilon\tilde{\Lambda}_2 = \begin{pmatrix} \varepsilon\tilde{\lambda}_{r+1} & & 0 \\ & \ddots & \\ 0 & & \varepsilon\tilde{\lambda}_H \end{pmatrix} = \begin{pmatrix} \lambda_{r+1} & & 0 \\ & \ddots & \\ 0 & & \lambda_H \end{pmatrix}, \tag{32}$$

$$\varepsilon\tilde{\Lambda}_3 = \begin{pmatrix} \varepsilon\tilde{\lambda}_{H+1} & & 0 \\ & \ddots & \\ 0 & & \varepsilon\tilde{\lambda}_L \end{pmatrix} = \begin{pmatrix} \lambda_{H+1} & & 0 \\ & \ddots & \\ 0 & & \lambda_L \end{pmatrix}. \tag{33}$$

Note that the normalized singular values $\tilde{\lambda}_j$ $(r+1 \leq j \leq L)$ are of the constant order.

The purpose of this subsection is to show that

$$\frac{d}{dt}\mathrm{E}_{gen} > 0 \tag{34}$$

if $r < H$ (overrealizable) and t satisfies

$$\frac{1}{\beta\tau^2\sqrt{N}(\tilde{\lambda}_H - \tilde{\lambda}_{H+1})} \ll t \ll \frac{\log\sqrt{N}}{\beta\tau^2\sqrt{N}(\tilde{\lambda}_H - \tilde{\lambda}_{H+1})}. \tag{35}$$

If this is proved, it means the existence of overtraining in overrealizable cases. Before entering into the details, we give an intuitive description of what happens in the learning process of a network (Fukumizu, 1998). The final state of the solution can be written by

$$W\Lambda^{\frac{1}{2}}\begin{pmatrix} I_H & 0 \\ 0 & 0 \end{pmatrix}\Lambda^{\frac{1}{2}}U^T. \tag{36}$$

This can be considered as the projection of Λ to the first H eigenspaces. We can deduce from eq.(22) that the solution is also approximately considered to be a projection of Λ to a H dimensional subspace, which converges to I_H exponentially. An important point is that the convergence speed of the first r components and the rest are different if $r < H$. The order of the former is roughly $O(e^{-aNt})$ and the latter is $O(e^{-b\sqrt{N}t})$ for some constant a and b. When the slow dynamics of the latter is the leading factor, the subspace of the projection is approximately

$$\Pi = \begin{pmatrix} I_r & 0 \\ 0 & I_{H-r} \\ 0 & O(e^{-\sqrt{N}t}) \end{pmatrix}. \tag{37}$$

It is easy to see the *shrinkage* of the components corresponding to $H - r$ redundant parameters in the projection $\Lambda^{-\frac{1}{2}}\Pi(\Pi^T\Pi)^{-1}\Pi^T\Lambda^{-\frac{1}{2}}$. This suppresses the effect of stochastic noise and makes a better estimation than the final one.

Let us start the theoretical verification with the singular value decompositions to simplify eq.(25). We have

$$S = \Phi\begin{pmatrix} \Lambda & 0 \\ 0 & -\Lambda \end{pmatrix}\Psi^T, \tag{38}$$

where $\Phi = \frac{1}{\sqrt{2}}\begin{pmatrix} U & U \\ W & -W \end{pmatrix}$. From the assumption (d), the singular value decomposition of $R(0)$ has the following form;

$$R(0) = \Theta J \Gamma G^T, \tag{39}$$

where

$$\Theta = \begin{pmatrix} P & 0 \\ 0 & Q \end{pmatrix}, \quad J = \begin{pmatrix} I_H \\ 0 \\ I_H \\ 0 \end{pmatrix}, \quad \text{and} \quad \Gamma = \begin{pmatrix} \gamma_1 & & 0 \\ & \ddots & \\ 0 & & \gamma_H \end{pmatrix}. \tag{40}$$

The matrixes P, Q and G are L, L, and H dimensional orthogonal matrixes respectively. Because of the assumption (e), Γ is invertible. Using these decompositions, RR^T can be

rewritten as

$$RR^T = \Phi\begin{pmatrix} e^{\beta\tau^2 N\Lambda t} & 0 \\ 0 & e^{-\beta\tau^2 N\Lambda t}\end{pmatrix}\Phi^T\Theta J \times \left[\Gamma^{-2} + \frac{1}{2}J^T\Theta^T\Phi\left\{\begin{pmatrix}\Lambda^{-1}e^{2\beta\tau^2 N\Lambda t} & 0 \\ 0 & -\Lambda^{-1}e^{-2\beta\tau^2 N\Lambda t}\end{pmatrix} - \begin{pmatrix}\Lambda^{-1} & 0 \\ 0 & -\Lambda^{-1}\end{pmatrix}\right\}\Phi^T\Theta J\right]^{-1} \times J^T\Theta^T\Phi\begin{pmatrix} e^{\beta\tau^2 N\Lambda t} & 0 \\ 0 & e^{-\beta\tau^2 N\Lambda t}\end{pmatrix}\Phi^T. \quad (41)$$

Let $C = (U^TP + W^TQ)$ and $D = (U^TP - W^TQ)$. Decompose these two matrixes as

$$C = \begin{pmatrix} C_H & * \\ C_3 & *\end{pmatrix}, \qquad D = \begin{pmatrix} D_H & * \\ D_3 & *\end{pmatrix}, \quad (42)$$

where $C_H, D_H \in M(H \times H; \mathbb{R})$, and $C_3, D_3 \in M((L-H) \times H; \mathbb{R})$. Then, we have

$$\begin{pmatrix} e^{\beta\tau^2 N\Lambda t} & 0 \\ 0 & e^{-\beta\tau^2 N\Lambda t}\end{pmatrix}\Phi^T\Theta J = \frac{1}{\sqrt{2}}\begin{pmatrix} e^{\beta\tau^2 N\Lambda_H t}C_H \\ e^{\beta\tau^2\sqrt{N}\tilde{\Lambda}_3 t}C_3 \\ e^{\beta\tau^2 N\Lambda_H t}D_H \\ e^{\beta\tau^2\sqrt{N}\tilde{\Lambda}_3 t}D_3\end{pmatrix}, \quad (43)$$

where $\Lambda_H = \begin{pmatrix}\Lambda_1 & 0 \\ 0 & \Lambda_2\end{pmatrix}$. Since $B(0) + FA(0)^T = Q\begin{pmatrix}\Gamma \\ O\end{pmatrix}G^T + W\Lambda U^T P\begin{pmatrix}\Gamma \\ O\end{pmatrix}G^T$, the assumption (f) assures that C_H is invertible. If we introduce

$$K = \begin{pmatrix} I_H \\ \Lambda_3^{-\frac{1}{2}}e^{\beta\tau^2\sqrt{N}\tilde{\Lambda}_3 t}C_3C_H^{-1}e^{-\beta\tau^2 N\Lambda_H t}\Lambda_H^{\frac{1}{2}} \\ \sqrt{-1}\Lambda_H^{-\frac{1}{2}}e^{-\beta\tau^2 N\Lambda_H t}D_HC_H^{-1}e^{-\beta\tau^2 N\Lambda_H t}\Lambda_H^{\frac{1}{2}} \\ \sqrt{-1}\Lambda_3^{-\frac{1}{2}}e^{-\beta\tau^2\sqrt{N}\tilde{\Lambda}_3 t}D_3C_H^{-1}e^{-\beta\tau^2 N\Lambda_H t}\Lambda_H^{\frac{1}{2}}\end{pmatrix}, \quad (44)$$

eq.(41) can be rewritten as

$$RR^T = 2\Phi\begin{pmatrix}\Lambda^{\frac{1}{2}} & 0 \\ 0 & \sqrt{-1}\Lambda^{\frac{1}{2}}\end{pmatrix}K\Psi^{-1}K^T\begin{pmatrix}\Lambda^{\frac{1}{2}} & 0 \\ 0 & \sqrt{-1}\Lambda^{\frac{1}{2}}\end{pmatrix}\Phi^T, \quad (45)$$

where

$$\begin{aligned}\Psi =& K^TK + \Lambda_H^{\frac{1}{2}}e^{-\beta\tau^2 N\Lambda_H t}C_H^T\Gamma^{-2}C_He^{-\beta\tau^2 N\Lambda_H t}\Lambda_H^{\frac{1}{2}} \\ &+ e^{-2\beta\tau^2 N\Lambda_H t} + \Lambda_H^{\frac{1}{2}}e^{-\beta\tau^2 N\Lambda_H t}C_H^{-1^T}C_3^T\Lambda_3^{-1}C_3C_H^{-1}e^{-\beta\tau^2 N\Lambda_H t}\Lambda_H^{\frac{1}{2}} \\ &- \Lambda_H^{\frac{1}{2}}e^{-\beta\tau^2 N\Lambda_H t}C_H^{-1^T}D_H^T\Lambda_H^{-1}D_HC_H^{-1}e^{-\beta\tau^2 N\Lambda_H t}\Lambda_H^{\frac{1}{2}} \\ &- \Lambda_H^{\frac{1}{2}}e^{-\beta\tau^2 N\Lambda_H t}C_H^{-1^T}D_3^T\Lambda_3^{-1}D_3C_H^{-1}e^{-\beta\tau^2 N\Lambda_H t}\Lambda_H^{\frac{1}{2}}. \end{aligned} \quad (46)$$

Note that $K\Psi^{-1}K^T$ is exponentially close to the orthogonal projection onto the subspace spanned by the first H components.

In the matrix K, the $(H+1,H)$-component has the slowest order in the convergence, and the order is $\exp\{-\beta\tau^2\sqrt{N}(\tilde{\lambda}_H - \tilde{\lambda}_{H+1})t\}$. Hereafter, we assume that there is a time interval such that the value of this component is very small but sufficiently larger than ε; that is,

$$\varepsilon \ll \exp\{-\beta\tau^2\sqrt{N}(\tilde{\lambda}_H - \tilde{\lambda}_{H+1})t\} \ll 1. \tag{47}$$

This is equivalent to the condition of eq.(35). We consider the main part of $K\Psi^{-1}K^T$, neglecting terms of faster convergence.

The matrix K can be further decomposed as

$$K = \begin{pmatrix} I_H \\ K_2 \\ \sqrt{-1}K_3 \\ \sqrt{-1}K_4 \end{pmatrix} = \begin{pmatrix} I_r & 0 \\ 0 & I_{H-r} \\ K_{21} & K_{22} \\ \sqrt{-1}K_{31} & \sqrt{-1}K_{32} \\ \sqrt{-1}K_{41} & \sqrt{-1}K_{42} \end{pmatrix}, \tag{48}$$

where $K_{21}, K_{41} \in M((L-H)\times r;\mathbb{R})$, $K_{22}, K_{42} \in M((L-H)\times(H-r))$, $K_{31} \in M(H\times r;\mathbb{R})$, and $K_{32} \in M(H\times(H-r))$. The convergence of K_{21}, K_3, and K_{41} is equal to or faster than the order of $\exp\{-\beta\tau^2 N(\lambda_r - \varepsilon\tilde{\lambda}_{r+1})t\}$. We further assume that

$$\exp\{-\beta\tau^2 N(\lambda_r - \varepsilon\tilde{\lambda}_{r+1})t\} \ll \varepsilon^{\frac{3}{2}} \exp\{-2\beta\tau^2\sqrt{N}(\tilde{\lambda}_H - \tilde{\lambda}_{H+1})t\} \tag{49}$$

holds in the interval of eq.(47). This assumption is naturally satisfied under the condition of eq.(35) if N is sufficiently large, since it is equivalent to

$$t \gg \frac{\frac{3}{2}\log\sqrt{N}}{\beta\tau^2 N\{\lambda_r + \frac{1}{\sqrt{N}}(\tilde{\lambda}_H - \tilde{\lambda}_{r+1} + \tilde{\lambda}_{H+1})\}}. \tag{50}$$

Hereafter, we use $\sim$ for an approximation up to the leading order of each component in matrixes. Then, the solution can be rewritten as

$$\begin{aligned} RR^T &\sim 2\Phi \begin{pmatrix} \Lambda^{\frac{1}{2}} & 0 \\ 0 & \sqrt{-1}\Lambda^{\frac{1}{2}} \end{pmatrix} K(I_H - K_2^T K_2)K^T \begin{pmatrix} \Lambda^{\frac{1}{2}} & 0 \\ 0 & \sqrt{-1}\Lambda^{\frac{1}{2}} \end{pmatrix} \Phi^T \\ &\sim 2\Phi \begin{pmatrix} \Lambda^{\frac{1}{2}} & 0 \\ 0 & \Lambda^{\frac{1}{2}} \end{pmatrix} \begin{pmatrix} I_H - K_2^T K_2 & K_2^T & -K_3^T & -K_4^T \\ K_2 & K_2 K_2^T & -K_2 K_3^T & -K_2 K_4^T \\ -K_3 & -K_3 K_2^T & K_3 K_3^T & K_3 K_4^T \\ -K_4 & -K_4 K_2^T & K_4 K_3^T & K_4 K_4^T \end{pmatrix} \begin{pmatrix} \Lambda^{\frac{1}{2}} & 0 \\ 0 & \Lambda^{\frac{1}{2}} \end{pmatrix} \Phi^T. \end{aligned} \tag{51}$$

Therefore, we obtain

$$
\begin{aligned}
A^TA &\sim U\Lambda^{\frac{1}{2}}\begin{pmatrix} I_H - K_2^TK_2 & K_2^T - K_4^T \\ K_2 - K_4 & K_2K_2^T \end{pmatrix}\Lambda^{\frac{1}{2}}U^T, \\
BA &\sim W\Lambda^{\frac{1}{2}}\begin{pmatrix} I_H - K_2^TK_2 & K_2^T + K_4^T \\ K_2 + K_4 & K_2K_2^T \end{pmatrix}\Lambda^{\frac{1}{2}}U^T, \\
BB^T &\sim W\Lambda^{\frac{1}{2}}\begin{pmatrix} I_H - K_2^TK_2 & K_2^T - K_4^T \\ K_2 + K_4 & K_2K_2^T \end{pmatrix}\Lambda^{\frac{1}{2}}W^T. \qquad (52)
\end{aligned}
$$

Now, we are in a position to prove overtraining. The first term in the right hand side of eq.(25) is written as

$$
\begin{aligned}
&\mathrm{Tr}\big[(F-BA)A^TA(BA-B_0A_0)^T\big] \\
&\quad\sim \mathrm{Tr}\Bigg[\Lambda^{\frac{1}{2}}\begin{pmatrix} K_2^TK_2 & -K_2^T - K_4^T \\ -K_2 + K_4 & I_{L-H} - K_2K_2^T \end{pmatrix}\Lambda\begin{pmatrix} I_H - K_2^TK_2 & K_2^T + K_4^T \\ K_2 - K_4 & K_2K_2^T \end{pmatrix}\Lambda^{\frac{1}{2}} \\
&\qquad\qquad \times\left\{\Lambda^{\frac{1}{2}}\begin{pmatrix} I_H - K_2^TK_2 & K_2^T + K_4^T \\ K_2 + K_4 & K_2K_2^T \end{pmatrix}\Lambda^{\frac{1}{2}} - W^TB_0A_0U\right\}\Bigg]. \qquad (53)
\end{aligned}
$$

Lemma 1 in Appendix and further decomposition of K_2 and K_4 show

$$
\begin{aligned}
&\mathrm{Tr}\big[(F-BA)A^TA(BA-B_0A_0)^T\big] \\
&\sim \mathrm{Tr}\Bigg[\begin{pmatrix} \Lambda_1^{\frac{1}{2}}K_{21}^TK_{21}\Lambda_1^{\frac{3}{2}} & \varepsilon^{\frac{3}{2}}\Lambda_1^{\frac{1}{2}}K_{21}^TK_{22}\tilde{\Lambda}_2^{\frac{3}{2}} & \varepsilon^{\frac{3}{2}}\Lambda_1^{\frac{1}{2}}K_{21}^TK_{22}\tilde{\Lambda}_2K_{22}^T\tilde{\Lambda}_2^{\frac{1}{2}} \\ \varepsilon^{\frac{1}{2}}\tilde{\Lambda}_2^{\frac{1}{2}}K_{22}^TK_{21}\Lambda_1^{\frac{3}{2}} & \varepsilon^2\tilde{\Lambda}_2^{\frac{1}{2}}(K_{22}^TK_{22}\tilde{\Lambda}_2 - K_{22}^T\tilde{\Lambda}_3K_{22})\tilde{\Lambda}_2^{\frac{1}{2}} & \varepsilon^2\tilde{\Lambda}_2^{\frac{1}{2}}K_{22}^TK_{22}\tilde{\Lambda}_2K_{22}^T\tilde{\Lambda}_3^{\frac{1}{2}} \\ -\varepsilon^{\frac{1}{2}}\tilde{\Lambda}_3^{\frac{1}{2}}K_{21}\Lambda_1^{\frac{3}{2}} & -\varepsilon^2\tilde{\Lambda}_3^{\frac{1}{2}}K_{22}\tilde{\Lambda}_2^{\frac{3}{2}} + \varepsilon^2\tilde{\Lambda}_3^{\frac{3}{2}}K_{22}\tilde{\Lambda}_2^{\frac{1}{2}} & \varepsilon^2\tilde{\Lambda}_3^{\frac{1}{2}}(-K_{22}\tilde{\Lambda}_2K_{22}^T + \tilde{\Lambda}_3K_{22}K_{22}^T)\tilde{\Lambda}_3^{\frac{1}{2}} \end{pmatrix} \\
&\quad\times\begin{pmatrix} O(\varepsilon) & -\varepsilon^{\frac{1}{2}}\Lambda_1^{\frac{1}{2}}K_{21}^TK_{22}\tilde{\Lambda}_2^{\frac{1}{2}} + O(\varepsilon) & \varepsilon^{\frac{1}{2}}\Lambda_1^{\frac{1}{2}}K_{21}^T\tilde{\Lambda}_3^{\frac{1}{2}} + O(\varepsilon) \\ -\varepsilon^{\frac{1}{2}}\tilde{\Lambda}_2^{\frac{1}{2}}K_{22}^TK_{21}\Lambda_1^{\frac{1}{2}} + O(\varepsilon) & \varepsilon\tilde{\Lambda}_2 - \varepsilon\tilde{\Lambda}_2^{\frac{1}{2}}K_{22}^TK_{22}\tilde{\Lambda}_2^{\frac{1}{2}} + O(\varepsilon^2) & \varepsilon\tilde{\Lambda}_2^{\frac{1}{2}}K_{22}^T\tilde{\Lambda}_3^{\frac{1}{2}} + O(\varepsilon^2) \\ \varepsilon^{\frac{1}{2}}\tilde{\Lambda}_3^{\frac{1}{2}}K_{21}\Lambda_1^{\frac{1}{2}} + O(\varepsilon) & \varepsilon\tilde{\Lambda}_3^{\frac{1}{2}}K_{22}\tilde{\Lambda}_2^{\frac{1}{2}} + O(\varepsilon^2) & \varepsilon\tilde{\Lambda}_3^{\frac{1}{2}}K_{22}K_{22}^T\tilde{\Lambda}_3^{\frac{1}{2}} + O(\varepsilon^2) \end{pmatrix}\Bigg]. \\
&\qquad\qquad (54)
\end{aligned}
$$

The leading order is $\varepsilon^3\exp\{-2\beta\tau^2\sqrt{N}(\tilde{\lambda}_H - \tilde{\lambda}_{H+1})t\}$. Note that the order $O(\varepsilon^4)\times\exp\{-\beta\tau^2\sqrt{N}(\tilde{\lambda}_H - \tilde{\lambda}_{H+1})t\}$, appeared in $(3,2)\times(2,3)$ part in eq.(54), is smaller under the condition of eq.(47), and the order $O(\varepsilon^{\frac{3}{2}})\times K_{21}$, appeared in the $(3,1)\times(1,3)$ part, is also smaller under the assumption of eq.(49). Thus, the main part is included in

$$
\begin{aligned}
&\mathrm{Tr}\Big[\varepsilon^3\tilde{\Lambda}_2^{\frac{1}{2}}K_{22}^TK_{22}\tilde{\Lambda}_2^{\frac{5}{2}} - \varepsilon^3\tilde{\Lambda}_2^{\frac{1}{2}}K_{22}^T\tilde{\Lambda}_3K_{22}\tilde{\Lambda}_2^{\frac{3}{2}} - \varepsilon^3\tilde{\Lambda}_3^{\frac{1}{2}}K_{22}\tilde{\Lambda}_2^2K_{22}^T\tilde{\Lambda}_3^{\frac{1}{2}} + \varepsilon^3\tilde{\Lambda}_3^{\frac{3}{2}}K_{22}\tilde{\Lambda}_2K_{22}^T\tilde{\Lambda}_3^{\frac{1}{2}}\Big] \\
&= \varepsilon^3\sum_{i=1}^{L-H}\sum_{j=1}^{H-r}\tilde{\lambda}_{r+j}(\tilde{\lambda}_{r+j} - \tilde{\lambda}_{H+i})^2\{(K_{22})_{ij}\}^2. \qquad (55)
\end{aligned}
$$

All the terms in eq.(55) are positive, and this proves the trace is strictly positive. It can be proved in almost the same way that the second term in eq.(25) is also strictly positive.

Thus, we obtain

$$\frac{d}{dt}\mathrm{E}_{gen} > 0 \tag{56}$$

if the target is overrealizable and t is in the interval of eq.(35).

From the above discussion, we can see that overtraining occurs in an intermediate interval, while overtraining of the average learning curve in Fig.4 seems to continue to the end. If we look at each trial for one set of training data, we find that some learning curves slightly decreases after showing overtraining. However, the effect of the decrease is very small, and the global figure of the curve is determined by the initial decrease and overtraining. It is also easy to see theoretically that this final decrease or increase is very small because it is caused by $O(e^{-aNt})$ terms.

5 Conclusion

We showed that strong overtraining is observed in batch learning of multilayer neural networks when the target is overrealizable. According to the experimental results on multilayer perceptrons and linear neural networks, this overtraining seems to be a universal property of multilayer models. We theoretically analyzed the batch training of linear neural networks, showed the existence of overtraining in an intermediate time interval in overrealizable cases.

Although the theoretical analysis in this paper is only on the linear neural network model, it is very suggestive to the phenomena of overtraining, which is observed in many application of multilayer neural networks. It will be extremely important to clarify this overtraining phenomena theoretically also in other models.

References

[1] Amari, S., Murata, N., & Müller, K. R. (1996). Statistical theory of overtraining – is cross-validation asymptotically effective? In D. S. Touretzky, M. C. Mozer, & M. E. Hasselmo (Eds.), *Advances in Neural Information Processing Systems 8*, (pp.176–182). Cambridge, MA: MIT Press.

[2] Baldi, P. & Chauvin, Y. (1991) Temporal evolution of generalization during learning in linear networks. *Neural Computation, 3*, 589–603.

[3] Baldi, P. F. & Hornik, K. (1995). Learning in linear neural networks: a survey. *IEEE Trans. Neural Networks, 6*(4), 837–858.

[4] Bartia, R. (1997). *Matrix Analysis*, New York: Springer-Verlag.

[5] Cramér, H. (1946). *Mathematical method of statistics*, (pp.497–506). Princeton, NJ: Princeton University Press.

[6] Fukumizu, K. (1996). A regularity condition of the information matrix of a multilayer perceptron network. *Neural Networks, 9*(5), 871–879.

[7] Fukumizu, K. (1997). Special statistical properties of neural network learning. In *Proc. 1997 International Symposium on Nonlinear Theory and Its Applications (NOLTA'97)*, (pp.747–750).

[8] Fukumizu, K. (1998). Effect of Batch Learning in Multilayer Neural Networks. In *Proc. 5th International Conference on Neural Information Processing (ICONIP'98)*, (pp.67–70).

[9] Helmke, U. & Moore, J. B. (1994). *Optimization and Dynamical Systems.* London: Springer-Verlag.

[10] Heskes, T. M. & Kappen, B. (1991). Learning process in neural networks. *Physical Review A, 44*(4), 2718–2726.

[11] Oja, E. (1989). A simplified neuron model as a principal component analyzer. *J. Math. Noiol., 15*, 267–273.

[12] Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986). Learning internal representations by error propagation. In D.E. Rumelhart, J.L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing*, Vol.1 (pp.318–362). Cambridge, MA: MIT Press.

[13] Sasagawa, T. (1982). On the finite escape phenomena for matrix Riccati equations. In *IEEE Trans. Automatic Control, 27*(4), 977-979.

[14] Yan, W., Helmke, U. & Moore, J. B. (1994). Global analysis of Oja's flow for neural networks. *IEEE Trans. Neural Networks, 5*(5), 674–683.

Appendix

A Perturbation of singular value decomposition

We prove the following

Lemma 1. *Let Z be a $L \times L$ matrix, and C_0 be a $L \times L$ matrix of the form;*

$$C_0 = \begin{pmatrix} C^{(0)} & 0 \\ 0 & 0 \end{pmatrix}, \tag{57}$$

where $C^{(0)}$ is a $r \times r$ matrix. For a sufficiently small positive number ε, we define $C_\varepsilon := C_0 + \varepsilon Z$. Let

$$C_\varepsilon = W \Sigma_\varepsilon U^T \tag{58}$$

be the singular value decomposition of C_ε, where W and U are orthogonal matrixes and Σ_ε is a diagonal matrix which has the singular values of C_ε. Then, the matrix $W^T C_0 U$ has the following form;

$$W^T C_0 U = \begin{pmatrix} C^{(0)} + O(\varepsilon) & O(\varepsilon) \\ O(\varepsilon) & O(\varepsilon^2) \end{pmatrix}. \tag{59}$$

Proof. It is known (Bartia, 1997, Section III) that Σ_ε has the form;

$$\Sigma_\varepsilon = \begin{pmatrix} \Sigma^{(0)} + O(\varepsilon) & 0 \\ 0 & O(\varepsilon) \end{pmatrix}, \tag{60}$$

where $\Sigma^{(0)}$ is the diagonal matrix of the singular values of $C^{(0)}$. For a $L \times L$ matrix A, we write $A = \left(\begin{smallmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{smallmatrix}\right)$, where A_{11} is a $r \times r$ matrix. Then, the upper-left block of eq.(58) leads

$$W_{11}^T(\Sigma^{(0)} + O(\varepsilon))U_{11} + W_{21}^T O(\varepsilon) U_{21} = C^{(0)} + \varepsilon Z_{11}. \tag{61}$$

This shows that W_{11} and U_{11} are of the constant order and

$$W_{11}^T \Sigma^{(0)} U_{11} = C^{(0)} + O(\varepsilon). \tag{62}$$

We know that W_{11} and U_{11} are invertible for a small ε. From the upper-right block of eq.(58), we have

$$W_{11}^T(\Sigma^{(0)} + O(\varepsilon))U_{12} + W_{21}^T O(\varepsilon) U_{22} = \varepsilon Z_{12}. \tag{63}$$

Therefore, U_{12} must be of the order $O(\varepsilon)$. Likewise, $W_{12} = O(\varepsilon)$. Now the assertion is straightforward. □